

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Sohm	Art Unit: 2193
Serial No.: 10/811,356	Confirmation No.: 2074
Filed: March 25, 2004	Examiner: Michael D. Yaary
	Docket: TI-35856

For: FAST FOURIER TRANSFORM OPERATION WITH REDUCED CACHE PENALTY

Appeal Brief under 37 C.F.R. §41.37

Board of Patent Appeals and Interferences
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This is Appellant's Appeal Brief filed pursuant to 37 C.F.R. §41.37 and the Notice of Appeal filed February 6, 2008 and further in response to the Notification of Non-Compliant Appeal Brief of April 16, 2008. The status of the claims has been corrected by stating that claims 6 to 9 and 11 are allowed.

TABLE OF CONTENTS

Section	Page
Real Party in interest	3
Related Appeals and Interferences	3
Status of Claims	3
Status of Amendments Filed After Final Rejection	3
Summary of Claimed Subject Matter	3
Grounds for Rejection to be Reviewed on Appeal	4
Arguments	5
Claims Appendix	13
Evidence Appendix	14
Related Proceedings Appendix	15

Real Party in Interest

The real party in interest in this application is Texas Instruments Incorporated, a corporation of Delaware with its principle place of business in Dallas, Texas. An assignment to Texas Instruments Incorporated is recorded at reel 014840 and frames 0005 and 0006.

Related Appeals and Interferences

There are no appeals of interferences related to this appeal in this application.

Status of the Claims

Claims 1, 3 and 4 are rejected and subject to this appeal. Claims 6 to 9 and 11 are allowed. Claims 2, 5 and 10 are canceled.

Status of Amendments Filed After Final Rejection

No amendments to the claims were proposed following the FINAL REJECTION of November 14, 2007.

Summary of Claimed Subject Matter

The subject matter of appealed independent claim 1 of this application is taught in the application as follows:

Claim 1	Disclosure
1. A method of performing a Fast Fourier Transform in a data processing apparatus having a data cache smaller than the data set of the Fast Fourier Transform, comprising the steps of:	Fast Fourier Transform: page 1, lines 8 and 9; page 5, lines 13 to 15; Figure 7, page 20, lines 21 to page 21, line 4: Figure 8, page 28, lines 1 to 6 data cache smaller than cache set: page 5, lines 17 to 22

dividing said input data into R continuous data sets where each of said R continuous data sets fit within the data cache;	dividing into R data sets: page 5, lines 28 to 30; page 29, lines 15 and 16 each R data set fits in data cache: page 23, line 17 to page 24, line 9
disposing said input data into memory, each R continuous data set in continuous memory locations with a space in memory locations from an end of one continuous data set to a beginning of a next continuous data set equal to the size of a cache line;	space between data sets of a cache line: page 5, line 30 to page 6, line 5; 930, Figure 9; page 29, line 13 to page 30, line 6
separately and independently performing a first stage radix-R butterfly computations on all the the R continuous data sets thereby producing R independent intermediate data sets each of which fits within the data cache; and	separately and independently performing a first stage radix-R butterfly computations: page 5, lines 17 to 22; page 6, lines 6 to 10; page 24, lines 9 to 17 and lines 24 and 25
successively performing second and all subsequent stage butterfly computations on each independent intermediate data set in turn producing corresponding output data.	performing second and subsequent stage butterfly computations: page 5, lines 22 to 27; page 6, lines 10 to 14; page 24, lines 13 to 19 and lines 25 and 26

Grounds for Rejection to be Reviewed on Appeal

Claims 1, 3 and 4 were rejected under 35 U.S.C. 103(a) as made obvious by the combination of Wadleigh U.S. Patent No. 6,088,714, Sayegh U.S. Patent No. 5,293,330 and Horton U.S. Patent No. 6,421,696.

Arguments

Claims 1, 3 and 4 were rejected under 35 U.S.C. 103(a) as made obvious by the combination of Wadleigh U.S. Patent No. 6,088,714, Sayegh U.S. Patent No. 5,293,330 and Horton U.S. Patent No. 6,421,696.

Claim 1 recites subject matter not made obvious by the combination of Wadleigh, Sayegh and Horton. Claim 1 recites "disposing said input data into memory, each R continuous data set in continuous memory locations with a space in memory locations from an end of one continuous data set to a beginning of a next continuous data set equal to the size of a cache line." The ADVISORY ACTION of February 1, 2008 at lines 6 to 11 of the continuation of paragraph 11 criticizes arguments in the response filed January 11, 2008 as attacking references individually where the rejection is based upon a combination of references. The FINAL REJECTION states at paragraph 8 on page 4:

"8. The combination of Wadleigh and Sayegh disclose disposing said input data into memory (Wadleigh column 4, lines 29-61) but the combination of Wadleigh and Sayegh do not disclose each R continuous data set in continuous memory locations with a space in memory locations from an end of one continuous data set to a beginning of a next continuous data set equal to the size of a cache line."

The arguments of the response filed January 11, 2008 were directed to the above quoted limitation of "disposing said input data into memory" which the Examiner stated was not found in Wadleigh or Sayegh. An argument that Horton fails to make obvious this limitation is proper, because 35 U.S.C. 103(a) requires the combination of references to make obvious each limitation of the claim and the FINAL REJECTION states that only Horton makes obvious this limitation. The arguments in the response filed January 11, 2008 were limited to the rejection presented by the Examiner. If

the Examiner now believes that some portion of Wadleigh or Sayegh makes obvious this "disposing said input data into memory," the Examiner is invited to point out these portions.

The FINAL REJECTION cites column 13, lines 34 to 63 of Horton as making obvious this limitation of "disposing said input data into memory." This teaching of Horton differs from the limitation of claim 1 in several aspects.

This limitation of claim 1 concerns the memory alignment of the input data ("disposing said input data into memory"). The cited portion of Horton clearly concerns the memory alignment of instructions. The portion of Horton cited in the FINAL REJECTION concerns the code fragment shown in column 13, lines 15 to 20 implementing the initialization of step 410 of Figure 7. The Applicant respectfully submits that it is known in the art that memory data alignment is not the same as memory instruction alignment. This application teaches that instructions are cached in a level 1 program cache 151 separate from the level 1 data cache 157 that caches data. Horton teaches that L1 instruction cache 214 and predocode cache 215 storing instructions are separate from L1 Dcache 226 storing data. Because both this application and Horton teach different caches for instructions and data, the instruction alignment taught Horton fails to make obvious the data alignment recited in claim 1. Accordingly, claim 1 is allowable over the combination of Wadleigh, Sayegh and Horton.

Claim 1 recites inserting memory spaces in differing locations and number than taught in Horton. Claim 1 recites "each R continuous data set in continuous memory locations with a space in memory locations from an end of one continuous data set to a beginning of a next continuous data set." This recitation thus requires R-1 such spaces in memory at locations corresponding to the size of the R continuous data sets. Horton states at column

13, lines 41 to 43 (within the portion cited in the FINAL REJECTION):

"when the starting byte of such an instruction occurs at the end of a cache line, i.e. in the last one, two, or, in some cases, three bytes of the cache line"

Column 13 makes clear at lines 40, 49 and 51 that "such an instruction" refers to instructions that are "short-decodable." Short-decodable instructions are apparently a subset of the instruction set of Horton. Thus Horton teaches the re-alignment takes place when a particular type instruction (short-decodable) has a particular relation to the end of a cache line (occurs within one, two or three bytes of the end of the cache line). The FINAL REJECTION points to no teaching of Horton that details the number or locations of the dummy instructions to be inserted. One skilled in the art would understand that dummy instruction insertion of Horton occurs at locations within the instruction code dependent upon the detail of instruction alignment relative to the cache lines. This would generally be variable in location. The FINAL REJECTION points to no portion of Horton as making obvious either the particular memory space between the R continuous data sets or the number of such spaces recited in claim 1. Accordingly, claim 1 is allowable over the combination of Wadleigh, Sayegh and Horton.

Claim 1 recites a differing amount of inserted memory space than taught in Horton. Claim 1 recites each memory space is "equal to the size of a cache line." Horton does not teach this amount of memory space. Horton states at column 13, lines 36 to 39:

"This mov instruction serves to displace the starting byte of the subsequent instruction 'lea ebx, [edi*2+edi]' from the end of one cache line to the beginning of the next cache line."

Horton also states at column 17, lines 14 to 17:

"Instruction (21) is inserted to properly align the subsequent instruction (22), i.e. to displace the starting byte of the subsequent instruction (22) away from an end region of a cache line to the beginning of the succeeding cache line."

This teaching of Horton at column 13, lines 41 to 43 would lead one skilled in the art to insert one, two or three bytes depending upon the relation of the alignment of the short-decodable instruction and the end of the cache line. This is a different displacement amount than recited in claim 1. The ADVISORY ACTION states at the continuation of paragraph 11, lines 17 to 19:

"Horton discloses spacing depending upon the number of bytes to be displaced (column 13, lines 63-65), thus the combination of Wadleigh, Sayegh, and Horton teach alignment anywhere from several bytes to an entire cache line, if necessary, depending on appropriate displacement."

Horton does not make obvious insertion of an entire cache line. Insertion of an entire cache line in Horton would not move the next instruction "away from an end region of a cache line to the beginning of the succeeding cache line" or achieve the purpose of displacing the "starting byte of the subsequent instruction (22) away from an end region of a cache line to the beginning of the succeeding cache line." Insertion of an entire cache line to shift a subsequent instruction to the beginning of a cache line would move the short-decodable instruction from the beginning of a cache line to the beginning of the next cache line. No dummy instruction insertion would be made in this case because the short-decodable instruction would already be at the beginning of a cache line and not within one, two or three bytes of the end of a cache line. Viewed another way, insertion of dummy instructions equal to a cache line would not change the relationship of a short-decodable instruction to the cache line boundary to better permit predecode

unit 212 to be able to determine the length of the instruction to be decoded (column 13, lines 43 and 44). An insertion of an entire cache line would have no advantage according to Horton. Horton would thus never provide dummy instruction insertion of an entire cache line as required by claim 1. Accordingly, claim 1 is allowable over the combination of Wadleigh, Sayegh and Horton.

Claim 1 teaches the purpose and effect of the claimed memory space differs from the teaching of Horton. This application teaches at page 29, line 13 to page 33, line 5 that using this memory space for alignment of data in memory avoids many cache conflict misses by assuring that the four input elements of a butterfly map to different cache sets. This application teaches that this mapping to different sets is a consequence of this memory alignment. Horton states at column 13, lines 39 to 53:

"Instructions such as the load effective address (lea) instruction normally invoke a short decode. However, when the starting byte of such an instruction occurs at the end of a cache line, i.e. in the last one, two, or, in some cases, three bytes of the cache line, predecode unit 212 is unable to determine the length of the instruction to be decoded. Thus, decode unit 220 cannot invoke a fast short decode and must dispatch the instruction to the much slower vector decode unit 256."

"Since it may be especially disadvantageous for the starting byte of a short-decodable instruction to be located at the last one, two, or three bytes of a cache line, instructions may be inserted prior to the short-decodable instruction to displace the short-decodable instruction so as to start in the next succeeding cache line."

This portion of Horton teaches the dummy instruction is inserted to speed decoding of the next instruction. Decoding is speeded by using a short decode when the instruction does not cross a cache line boundary rather than requiring vector decode unit 256 when the instruction crosses a cache line boundary. The Applicant submits that this motivation for aligning instructions taught in Horton so

differs from the motivation to reduce cache misses taught in this application that one skilled in the art would not be motivated to adopt the instruction alignment teaching of Horton in combination with the teachings of Wadleigh and Sayegh to align data as recited in claim 1. Accordingly, claim 1 is allowable over the combination of Wadleigh, Sayegh and Horton.

The ADVISORY ACTION states at the continuation of paragraph 11 at lines 20 to 22:

"In response to applicant's argument that the purpose in Horton differs than that of the instant claim 1, the fact that applicant has recognized another advantage which would flow naturally following the suggestion of the prior art cannot be the basis for patentability when the differences would otherwise be obvious. See *Ex parte Obiaya*, 227 USPQ 58, 60 (Bd. Pat. App. & Inter. 1985)."

The Examiners argument regarding finding "another advantage" of the teachings of the prior art are correct. However, Horton does not teach any of the substantive limitations of the "disposing said input data into memory" recited in claim 1 such as location and length of the insertion. Thus the difference in purpose between claim 1 and the teachings of Horton is additional evidence that the difference between aligning instructions and aligning data, the differing number and locations of the insertions and the differing amount of the insertions are such that Horton fails to make obvious claim 1. Accordingly, claim 1 is allowable over the combination of Wadleigh, Sayegh and Horton.

The differences between the requirements of claim 1 and the teachings of Horton regarding this limitation are summarized in the following table.

	Claim 1	Horton
Type	Data "disposing said input data into memory"	Instructions "the above code sequence" column 13, lines 35 and 36
Location and Number	Fixed "each R continuous data set in continuous memory locations with a space in memory locations from an end of one continuous data set to a beginning of a next continuous data"	Variable "when the starting byte of such an instruction occurs at the end of a cache line, i.e. in the last one, two, or, in some cases, three bytes of the cache line" column 13, lines 41 to 43
Amount	Fixed "equal to the size of a cache line"	Variable "Depending on the number of bytes to be displaced, various short-encoded instructions may be inserted to generate the displacement." column 13, lines 64 to 66
Purpose	Avoid cache conflict misses "These conflict misses can be avoided in this radix-4 example if each of the four input elements of a butterfly maps to a different set. To achieve this, the input data is split into four sets consisting of N/4 complex samples each and a gap of one cache line is inserted after each set." Application at page 29, lines 13 to 17	Speed instruction decode "Although it may seem counter-intuitive, by strategically adding a few such instructions in a program in appropriate locations, the speed of execution and throughput of the program as a whole may be significantly increased." column 13, lines 59 to 63

In conclusion, Horton teaches: (1) alignment of instructions rather than alignment of data as recited in claim 1; (2) insertion of dummy instructions at in differing number and differing locations, variable dependent upon alignment of short-decodable instructions to cache line boundaries, rather than the fixed number

and locations between each R continuous data sets as recited in claim 1; (3) insertion of one, two or three bytes to move the following instruction from the end region of one cache line to the beginning of the next cache line rather than the memory space equal to a cache line recited in claim 1; and (4) a different purpose for the alignment, speeding instruction decode rather than reducing the number of cache misses as taught in this application. As a consequence of these many differences, Horton fails to make obvious this limitation of claim 1. The FINAL REJECTION does not allege that Wadleigh, Sayegh or their combination teaches this limitation. Accordingly, claim 1 is allowable over the combination of Wadleigh, Sayegh and Horton.

Claims 3 and 4 are allowable by dependent upon allowable claim 1.

If the Examiner has any questions or other correspondence regarding this application, Applicants request that the Examiner contact Applicants' attorney at the below listed telephone number and address to facilitate prosecution.

Texas Instruments Incorporated
P.O. Box 655474 M/S 3999
Dallas, Texas 75265
(972) 917-5290
Fax: (972) 917-4418

Respectfully submitted,

/Robert D. Marshall, Jr./
Robert D. Marshall, Jr.
Reg. No. 28,527

CLAIMS APPENDIX

1 1. A method of performing a Fast Fourier Transform in a data
2 processing apparatus having a data cache smaller than the data set
3 of the Fast Fourier Transform, comprising the steps of:

4 dividing said input data into R continuous data sets where
5 each of said R continuous data sets fit within the data cache;

6 disposing said input data into memory, each R continuous data
7 set in continuous memory locations with a space in memory locations
8 from an end of one continuous data set to a beginning of a next
9 continuous data set equal to the size of a cache line;

10 separately and independently performing a first stage radix-R
11 butterfly computations on all the the R continuous data sets
12 thereby producing R independent intermediate data sets each of
13 which fits within the data cache; and

14 successively performing second and all subsequent stage
15 butterfly computations on each independent intermediate data set in
16 turn producing corresponding output data.

1 3. The method of claim 1, wherein:

2 said radix-R is radix-2.

1 4. The method of claim 1, wherein:

2 said radix-R is radix-4.

Evidence Appendix

None

Related Proceedings Appendix

None